

## CLAIMS

1        1. In an intermediate network device having at least one line card defining a plu-  
2        rality of ports for receiving and forwarding messages and two or more supervisors, each  
3        supervisor configured to run one or more applications to facilitate message handling by  
4        the network device, a method for continuing operation of at least one application despite  
5        crashes or failures, the method comprising the steps of:

6                designating a first supervisor to be an active supervisor and a second supervisor to  
7        be a standby supervisor for the network device;  
8                executing the at least one application at the active supervisor;  
9                holding the at least one application at the standby supervisor in a dormant state;  
10                transmitting state information generated during execution of the at least one ap-  
11        plication from the active supervisor to the standby supervisor;  
12                storing the state information at the standby supervisor; and  
13                in response to a failure at the active supervisor, carrying on execution of the at  
14        least one application at the standby supervisor based upon at least some of the stored state  
15        information.

1        2. The method of claim 1 further comprising the step of defining a synchroniza-  
2        tion database having one or more synchronization records at the active supervisor,  
3        wherein

4                the synchronization records store state information to be transmitted to the  
5        standby supervisor.

1        3. The method of claim 2 further comprising the step of updating one or more of  
2        the synchronization records in response to an operating change at the at least one appli-  
3        cation program.

1        4. The method of claim 3 wherein the transmitting step comprises the step of  
2        sending the one or more updated synchronization records to the standby supervisor.

1       5. The method of claim 1 further comprising the steps of:  
2           generating a sequence number for use in instructing the at least one line card to  
3           change operating condition;  
4           sending the sequence number to the at least one line card with the instruction; and  
5           storing the sequence number at the at least one line card.

1       6. The method of claim 5 further comprising the steps of:  
2           sending the sequence number to the standby supervisor;  
3           storing the sequence number at the standby supervisor; and  
4           in response to a failure of the active supervisor, comparing the sequence number  
5           stored at the standby supervisor with the sequence number at the at least one line card.

1       7. The method of claim 6 further comprising the step of continuing operation of  
2           the at least one line card, following a crash or failure of the active supervisor, if the se-  
3           quence number stored at the at least one line card is one of (a) less than or equal to or (b)  
4           greater than the sequence number stored at the standby supervisor.

1       8. The method of claim 7 further comprising the step of resetting the at least one  
2           line card, following a crash or failure of the active supervisor, if the sequence number  
3           stored at the at least one line card is one of (a) greater than or (b) less than or equal to the  
4           sequence number stored at the standby supervisor.

1       9. The method of claim 1 further comprising the steps of:  
2           determining the validity of the state information stored at the standby supervisor  
3           following a crash or failure of the active supervisor; and  
4           blocking the at least one application from utilizing state information determined  
5           to be invalid in its execution.

1       10. The method of claim 1 further comprising the steps of:  
2           creating, at the active supervisor, an instance of an event in response to a request  
3           from an application;

4 providing the event instance to the requesting application for processing;  
5 providing the event instance to any listening applications that have registered for  
6 the event for processing;  
7 passing the event instance to the standby supervisor;  
8 receiving notifications from the requesting and listening applications that they  
9 have completed their processing of the event instance;  
10 passing the notifications to the standby supervisor; and  
11 in response to receiving notifications from the requesting and all listening appli-  
12 cations, closing the event instance at the active and standby supervisors.

1 11. The method of claim 10 further comprising the step of:  
2 in response to a crash or failure of the active supervisor, determining whether one  
3 or more event instances passed to the standby supervisor remain open;  
4 for a given event instance that remains open, identifying the requesting and lis-  
5 tening applications that have not completed their processing of the given event instance;  
6 for each requesting and listening application that has not completed its processing  
7 of the given event instance, calling a recovery function defined by the respective applica-  
8 tion to handle the open event instance.

1 12. An intermediate network device for use in a computer network, the network  
2 device comprising:

3 a first supervisor card in communicating relationship with the one or more line  
4 cards;

5 a second supervisor card in communication relations with the first supervisor  
6 card;

7 an application loaded onto the first and second supervisor cards, the application  
8 configured to define and manipulate a plurality of state variables; and

9 a high availability entity disposed on both the first and second supervisor cards,  
10 the high availability entities comprising:

11 an event mechanism for notifying a selected one of the first or second su-  
12 pervisor cards of changes to the application's state variables; and

a database mechanism for storing the state variables at the first and second supervisor cards.

1        13. The network device of claim 12 wherein:  
2            the first supervisor card is designated as an active supervisor card and the second  
3            supervisor card is designated as a standby supervisor card;  
4            the application is allowed to run on the active supervisor card but not on the  
5            standby supervisor card; and  
6            in response to a crash or failure of the active supervisor card, the application car-  
7            ries on its execution from the standby supervisor card utilizing at least some of the state  
8            variables stored at the database mechanism of the standby supervisor card.

1        14. The network device of claim 12 further comprising at least one line card de-  
2            fining a plurality of ports for forwarding messages across the computer network, the at-  
3            least one line card in communicating relationship with the first and second supervisor  
4            cards and configured to receive and maintain port state information from the application,  
5            wherein  
6            the high availability entities at the first and second supervisor cards further com-  
7            prise:  
8            a sequence mechanism for ensuring that the state variables stored at the  
9            first and second supervisor cards are consistent with the port state information  
10            maintained at the at least one line card.